Homework questions, week 4

Econ 103

1 Daily Homework questions

The questions in bold font are due on **Wednesday 15th June**. You do not need to hand in the questions that are not in bold, though these will be useful to complete for your own understanding.

Lecture 12: Sampling Distributions and Estimation I

Textbook questions:
Chapter 6: **1, 5, 7**

Additional questions: none

Lecture 13: Sampling Distributions and Estimation II

Textbook questions:
Chapter 7: 1, **9**, 13, **17**, **18**, 19

Additional questions: none

The answer in the back of the book for 7-19 is wrong. *I will provide full solutions to 7-13 since it's hard, 7-18 since it's even-numbered, and 7-19 since the book is wrong.*

Solution: 7-13
The point is that S, the number of successes in n trials each with probability π of success, is a Binomial(n, π) random variable. We calculated the mean and variance of such a RV in class (see the slides) and we will use this information to find the MSE of $P = S/n$ as well as that of

$$P^* = \frac{nP + 1}{n + 2} = \left(\frac{n}{n+2}\right) P + \left(\frac{1}{n+2}\right)$$

The reason the book gives you the above expression is to give you a hint: namely that once you've solved for the MSE of P you can use this to get the MSE of P^* fairly easily by writing it as above.

$$\begin{aligned}
E[P] &= E[S/n] = E[S]/n = n\pi/n = \pi \\
\text{Bias}(P) &= E[P] - \pi = \pi - \pi = 0 \\
Var(P) &= Var(S/n) = Var(S)/n^2 = n\pi(1-\pi)/n^2 = \pi(1-\pi)/n \\
MSE(P) &= \text{Bias}(P)^2 + Var(P) = 0^2 + \pi(1-\pi)/n = \pi(1-\pi)/n
\end{aligned}$$

where we have used our rules for manipulating expectation and variance, as well as the expressions for the mean and variance of a Binomial random variable. Now:

$$\begin{aligned}
E[P^*] &= E\left[\left(\frac{n}{n+2}\right)P + \left(\frac{1}{n+2}\right)\right] = \left(\frac{n}{n+2}\right)E[P] + \left(\frac{1}{n+2}\right) \\
&= \left(\frac{n}{n+2}\right)\pi + \left(\frac{1}{n+2}\right) \\
\text{Bias}(P^*) &= E[P^*] - \pi = \left(\frac{n}{n+2}\right)\pi + \left(\frac{1}{n+2}\right) - \pi \\
&= \left(\frac{n}{n+2} - 1\right)\pi + \left(\frac{1}{n+2}\right) = \frac{1-2\pi}{n+2} \\
Var(P^*) &= Var\left[\left(\frac{n}{n+2}\right)P + \left(\frac{1}{n+2}\right)\right] = \left(\frac{n}{n+2}\right)^2 Var(P) \\
&= \frac{n^2}{(n+2)^2}\frac{\pi(1-\pi)}{n} = \frac{n\pi(1-\pi)}{(n+2)^2} \\
MSE(P^*) &= \text{Bias}(P^*)^2 + Var(P^*) = \left(\frac{1-2\pi}{n+2}\right)^2 + \frac{n\pi(1-\pi)}{(n+2)^2} \\
&= \frac{(1-2\pi)^2 + n\pi(1-\pi)}{(n+2)^2} = \frac{1 - 4\pi + 4\pi^2 + n\pi - n\pi^2}{(n+2)^2} \\
&= \frac{1 + (n-4)\pi - (n-4)\pi^2}{(n+2)^2} = \frac{1 + \pi(1-\pi)(n-4)}{(n+2)^2}
\end{aligned}$$

If we take limits, we'll see that both P and P^* are consistent, since their mean-squared errors go to zero as $n \to \infty$. For different values of π and n, however, the two estimators will have different MSE. Parts (d) and (e) of this question simply ask you to plug in various values and compare.

Solution: 7-18

The point of this question is non-response bias: the people who respond are not representative of the population as a whole. Note that P and P^* as defined in this question *do not correspond* to question 7-13. Our goal is to estimate the population proportion who will buy a computer. Using the table, we calculate the total number of people who will buy a computer as:

$$0.2 \times 40 + 0.04 \times 5 + 0.1 \times 3 + 0.2 \times 2 = 1.7 \text{ million}$$

which corresponds to a fraction $\pi^* = 1.7/50 = 0.034$. Again using the table, the number of people who will buy a computer *among the sub-population who would respond* can be calculated as:

$$0.2 \times 7 + 0.04 \times 1 + 0.1 \times 1 + 0.2 \times 1 = 0.48 \text{ million}$$

which corresponds to a fraction $\pi = 0.48/10 = 0.048$. The point is that $\pi \neq \pi^*$. In other words, the proportion of people who would buy a computer *differs* across people who would and would not respond to the phone survey. The estimator P is based on calling 1000 people chosen at random and recording responses for *only those who reply*. The proportion of people who will reply is $10/50 = 1/5$. Thus, P will end up with a sample size of approximately $n = 200$ individuals. These individuals correspond to the sub-population for which the proportion who would buy a computer is $\pi^* = 0.048$. In contrast, the estimator P^* is based on calling $n^* = 100$ people chosen at random and then following up with these people repeatedly until *all of them respond*. Thus, P^* draws from the *full population*, in which a proportion $\pi^* = 0.034$ of people will buy a computer. The *true parameter* is π^* since we want to estimate the *overall* fraction of people who will buy a computer, *not* the fraction of people who would buy a computer among those who are likely to respond to a telephone survey. Hence bias is calculated *relative to* π^*. Variance is calculated *relative to the mean of each sampling distribution*. For P this mean is π while for P^* it is π^*. That is:

$$\begin{aligned}
MSE(P) &= \text{Bias}(P)^2 + Var(P) = (E[P] - \pi^*)^2 + E[(P-\pi)^2] \\
&= (\pi - \pi^*)^2 + E[(P-\pi)^2] \\
&= (\pi - \pi^*)^2 + \pi(1-\pi)/n \\
MSE(P^*) &= \text{Bias}(P^*)^2 + Var(P^*) = (\pi^* - \pi^*)^2 + E[(P^* - \pi^*)^2] \\
&= E[(P^* - \pi^*)^2] = \pi^*(1-\pi^*)/n^*
\end{aligned}$$

The estimator P^* does not have any bias because of the follow-ups to ensure that everyone in the original random sample responds. However, since it is based on a

smaller sample, we would expect it to have a higher variance. The question is how this trade-off comes out in the expressions for MSE. To find out, we simply plug in the values $\pi^* = 0.034$, $\pi = 0.048$, $n = 200$ and $n^* = 100$. We find $MSE(P^*) \approx 0.000328$ and $MSE(P) \approx 0.000424$.

Solution: 7-19 *THE ANSWER IN THE BOOK IS WRONG!*

Specifically, they take $n = 100$ rather than $n = 200$ when calculating the variance of P. The reason this is wrong is because 1000 is the number of people *called* not the number of people who *reply*. The question statement specifically states that P should be calculated relative to those who respond.

All we have to do in this question is take the square root of the answers from the previous question. We find that $RMSE(P^*) \approx 0.018$ and $RMSE(P) \approx 0.021$. These are the root mean squared errors for estimators of the population *proportion*. To answer the question for estimators of *market size*, i.e. the population proportion multiplied by the size of the market (50 million), we simply multiply each of the RMSE values by 50 million yielding values of approximately 900,000 and 1,000,000 respectively.

Lecture 14: Confidence Intervals I

Textbook questions:
Chapter 8: **1**, 3, 5, **7**

Additional questions:

1. **For this question assume that we have a random sample from a normal distribution with unknown mean but *known* variance.**
   (a) Suppose that we have 36 observations, the sample mean is 5, and the population variance is 9. Construct a 95% confidence interval for the population mean.

   Solution: Since the population variance is 9, the population standard deviation is 3. Hence, the desired confidence interval is $5 \pm 2 \times 3/\sqrt{36} = 5 \pm 1 = (4, 6)$

   (b) Repeat the preceding with a population variance of 25 rather than 9.

Solution: The population standard deviation becomes 5 so the confidence interval becomes $5 \pm 2 \times 5/\sqrt{36} = 5 \pm 10/6 \approx (3.3, 6.7)$

(c) Repeat the preceding with a sample size of 25 rather than 36.

Solution: $5 \pm 2 \times 5/\sqrt{25} = 5 \pm 2 = (3, 7)$

(d) Repeat the preceding but construct a 50% rather than 95% confidence interval.

Solution: Here we need to use R to get the appropriate quantile:

```
SE <- 5/sqrt(25)
ME <- qnorm(1 - 0.5/2) * SE
Lower <- 5 - ME
Upper <- 5 + ME
c(Lower, Upper)
## [1] 4.326 5.674
```

(e) Repeat the preceding but construct a 99% rather than a 50% confidence interval.

Solution: Again we use R to get the appropriate quantile:

```
SE <- 5/sqrt(25)
ME <- qnorm(1 - 0.01/2) * SE
Lower <- 5 - ME
Upper <- 5 + ME
c(Lower, Upper)
## [1] 2.424 7.576
```

Lecture 15: Confidence Intervals II

Textbook questions: none

Additional questions:

1. **All other things equal, how would the following change the width of a confidence interval for the mean of a normal population? Explain.**

Solution: All answers below are based on the following expression for a $(1-\alpha) \times 100\%$ confidence interval for the mean of a normal population when the population standard deviation is unknown:

$$\bar{X}_n \pm \texttt{qt}(1-\alpha/2, df = n-1) \times \frac{S}{\sqrt{n}}$$

The width of this interval is

$$\text{Width} = 2 \times \texttt{qt}(1-\alpha/2, df = n-1) \times \frac{S}{\sqrt{n}}$$

(a) The sample mean is smaller.

Solution: No effect: width doesn't involve the sample mean.

(b) The population mean is smaller.

Solution: No effect: width doesn't involve the population mean.

(c) The sample standard deviation is smaller.

Solution: If S decreases, all other things constant, the width decreases.

(d) The sample size is smaller.

Solution: Changing sample size has two effects but they both go in the same direction. First, if n gets smaller, $\texttt{qt}(1-\alpha/2, df = n-1)$ gets larger as we can see from the table presented in the lecture slides. Second, as n gets smaller holding all other things fixed, $S/\sqrt{n}$ gets larger. Hence, decreasing sample size, all other things equal, increases the width.

2. **In this question you will carry out a simulation exercise similar to the one I used to make the plot of twenty confidence intervals from lecture 14. R Tutorial 4 will be useful for this question - I suggest you complete this before you do this question. To hand in this question, please email me a file with your R code in it (send your R script file).**

(a) Write a function called `my.CI` that calculates a confidence interval for the mean of a normal population when the population standard deviation is known. It should take three arguments: `data` is a vector containing the observed data from which

we will calculate the sample mean, `pop.sd` is the population standard deviation, and `alpha` controls the confidence level (e.g. `alpha = 0.1` for a 90% confidence interval). Your function should return a vector whose first element is the lower confidence limit and whose second element is the upper confidence limit. Test out your function on a simple example to make sure it's working properly.

Solution:

```
my.CI <- function(data, pop.sd, alpha){

  x.bar <- mean(data)
  n <- length(data)

  SE <- pop.sd / sqrt(n)
  ME <- qnorm(1 - alpha/2) * SE

  lower <- x.bar - ME
  upper <- x.bar + ME

  out <- c(lower, upper)
  return(out)

}
```

Testing this out on fake data containing twenty-five zeros and assuming a population variance of one, we have

```
fake.data <- rep(0, 25)
my.CI(fake.data, pop.sd = 1, alpha = 0.05)
## [1] -0.392  0.392
```

If we calculated the corresponding interval by hand, assuming a population standard deviation of one, we'd get

$$0 \pm 2 \times 1 \times 1/5 = (-0.4, 0.4)$$

Which is almost exactly the same. The reason for the slight discrepancy is that when working by hand we use the approximation `qnorm(0.975)`≈ 2 whereas the exact value, which R provides, is more like 1.96.

(b) Write a function called `CI.sim` that takes a single argument `sample.size`. Your function should carry out the following steps. First generate `sample.size` draws from a standard normal distribution. Second, pass your sample of standard normals

to `my.CI` with `alpha` set to 0.05 and `pop.sd` set to 1. Third, return the resulting confidence interval. Test your function on a sample of size 10. (What we're doing here is constructing a 95% confidence interval for the mean of a normal population using simulated data. The population mean is in fact zero, but we want to see how our confidence interval procedure works. To do this we "pretend" that we don't know the population mean and only know the population variance. Think about this carefully and make sure you understand the intuition.)

Solution:

```
CI.sim <- function(sample.size){
    sims <- rnorm(sample.size)
    CI <- my.CI(sims, pop.sd = 1, alpha = 0.05)
    return(CI)
}
CI.sim(10)
## [1] 0.01118 1.25077
```

(c) Use `replicate` to construct 10000 confidence intervals based on simulated data using the function `CI.sim` with `sample.size` equal to 10. (Note that `replicate` will, in this case, return a matrix with *2* rows and *10000* columns. Each column corresponds to one of the simulated confidence intervals. The first row contains the lower confidence limit while the second row contains the upper confidence limit.) Calculate the proportion of the resulting confidence intervals contain the true population mean. Did you get the answer you were expecting?

Solution:

```
simCIs <- replicate(10000, CI.sim(10))
simCIs[,1:5]
##          [,1]    [,2]    [,3]    [,4]     [,5]
## [1,] -1.17035 -1.0628 -0.7007 -0.2710 -1.22072
## [2,]  0.06924  0.1768  0.5389  0.9686  0.01887
lower <- simCIs[1,]
upper <- simCIs[2,]
covers.truth <- (lower < 0) & (upper > 0)
sum(covers.truth)/length(covers.truth)
## [1] 0.9513
```

The answer is pretty much dead on: almost exactly 95% of the intervals contain

the true population mean (zero).

(d) Repeat the preceding but rather than using `CI.sim` write a new function called `CI.sim2`. This new function should be identical to `CI.sim` except that, when calling `my.CI`, it sets `pop.sd = 1/2` rather than 1. How do your results change? Try to provide some intuition for any differences you find.

Solution:

```
CI.sim2 <- function(sample.size){
    sims <- rnorm(sample.size)
    CI <- my.CI(sims, pop.sd = 1/2, alpha = 0.05)
    return(CI)
}
simCIs <- replicate(10000, CI.sim2(10))
simCIs[,1:5]
##        [,1]     [,2]    [,3]     [,4]    [,5]
## [1,] -1.141 -0.52405 -0.1795 -0.06908 -0.4866
## [2,] -0.521  0.09575  0.4403  0.55071  0.1331
lower <- simCIs[1,]
upper <- simCIs[2,]
covers.truth <- (lower < 0) & (upper > 0)
sum(covers.truth)/length(covers.truth)
## [1] 0.6761
```

In this case the procedure didn't work: many fewer than 95% of the intervals contain the true population mean. The problem is that `CI.sim2` constructs a confidence interval using the *wrong* population standard deviation! Since it uses 1/2 rather than 1, the resulting intervals are too short, so too few of them contain the true population mean.

2 R Tutorials

You should complete R Tutorial #4 by **Thursday 16th June**.

R tutorials will be posted in Piazza, with solution code.